

# **NMPG: Network-Based Multiplayer PingPong Game**

Team 1: 4 to 6

21800426 안예온

21800436 양희찬

21800476 유현도

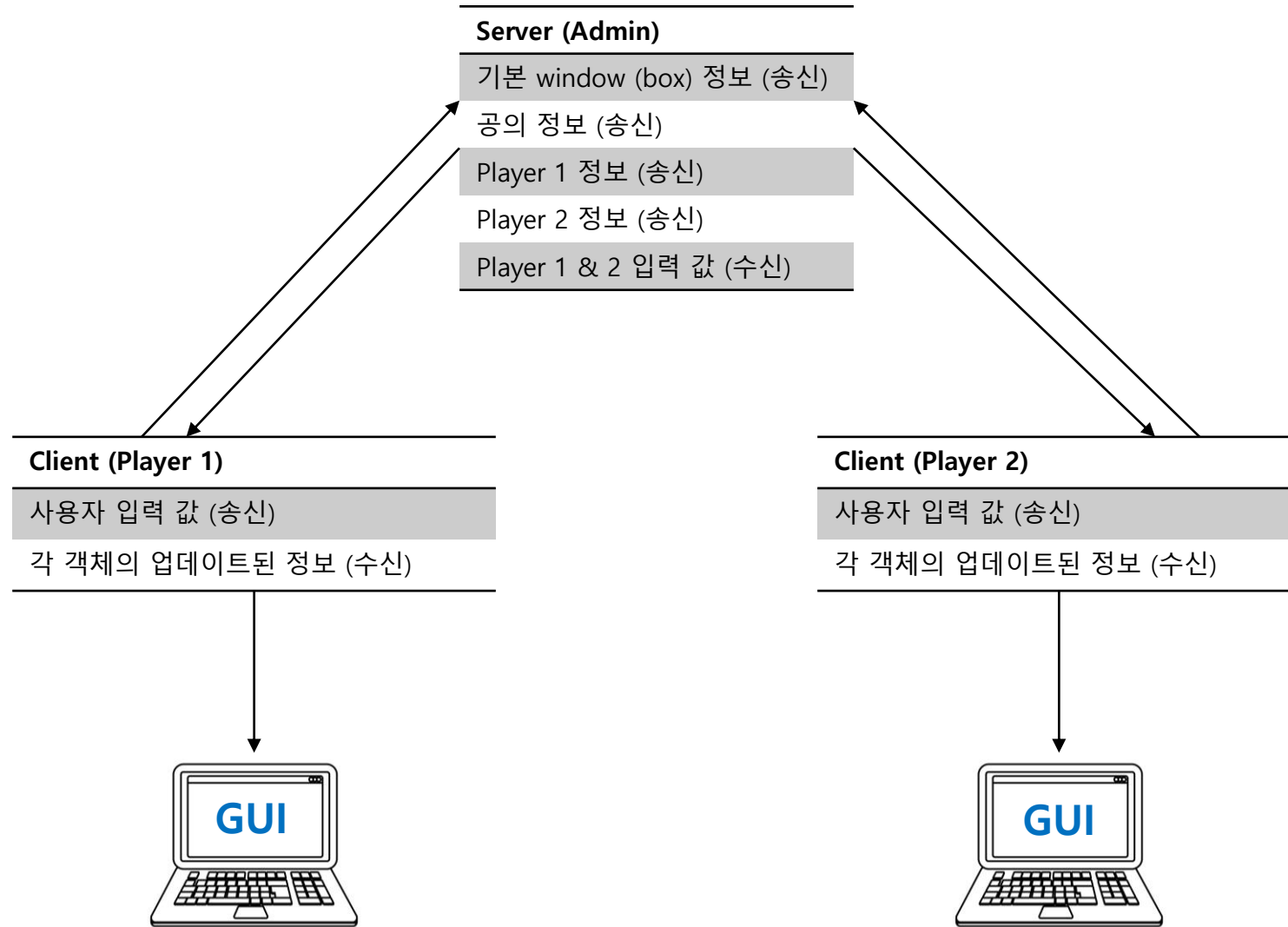
# 팀원

- 21800426 안예온 / AI 컴퓨터공학심화
- 21800436 양희찬 / AI 컴퓨터공학심화
- 21800476 유현도 / 전자, 전산

# 개발 목표

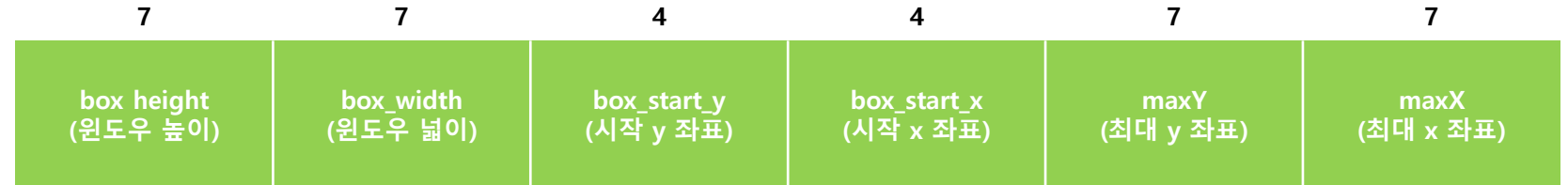
- 2명의 사용자가 개인 기기로 서버에 접속해서 delay 없이 함께 즐길 수 있는 핑퐁 게임 개발:
  - 사용자측:
    - 키보드 **입력 값**을 서버로 전송
    - 서버로부터 **기본 정보**와 각 **객체들의 위치** 정보를 받아 **출력**한다
  - 서버측:
    - 사용자로부터 **입력 값**을 받아, 이에 따라 정보를 **업데이트** 한다
    - 사용자에게 업데이트된 **정보**를 전송한다

# 시스템 구조

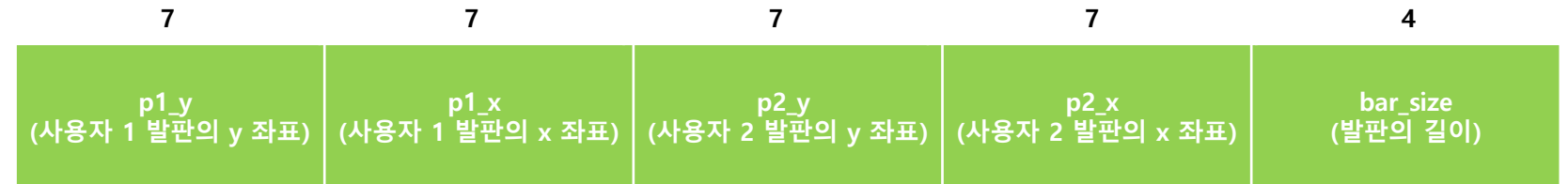


# 설계 프로토콜: TCP

- Window (Box) 정보



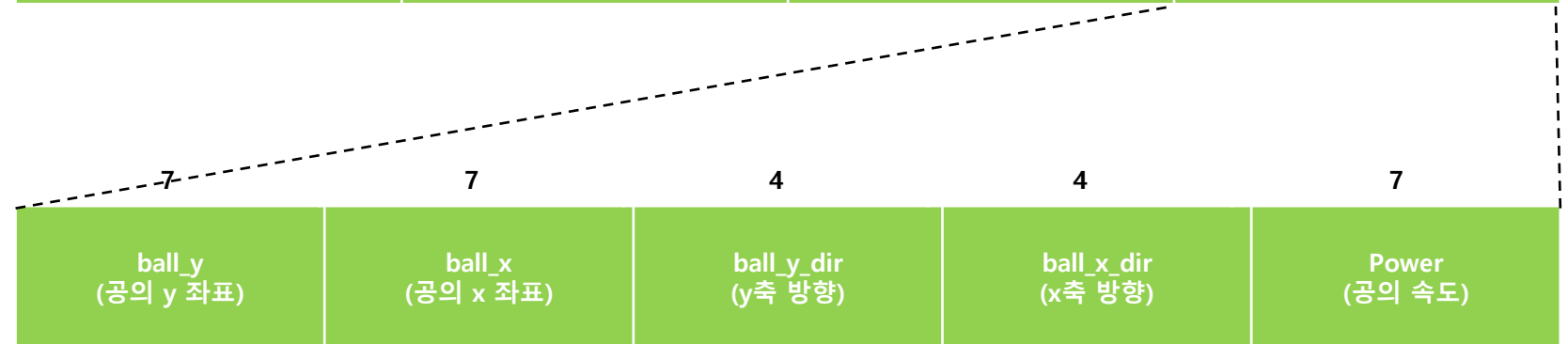
- 사용자 (1 & 2) 정보



- 게임 정보

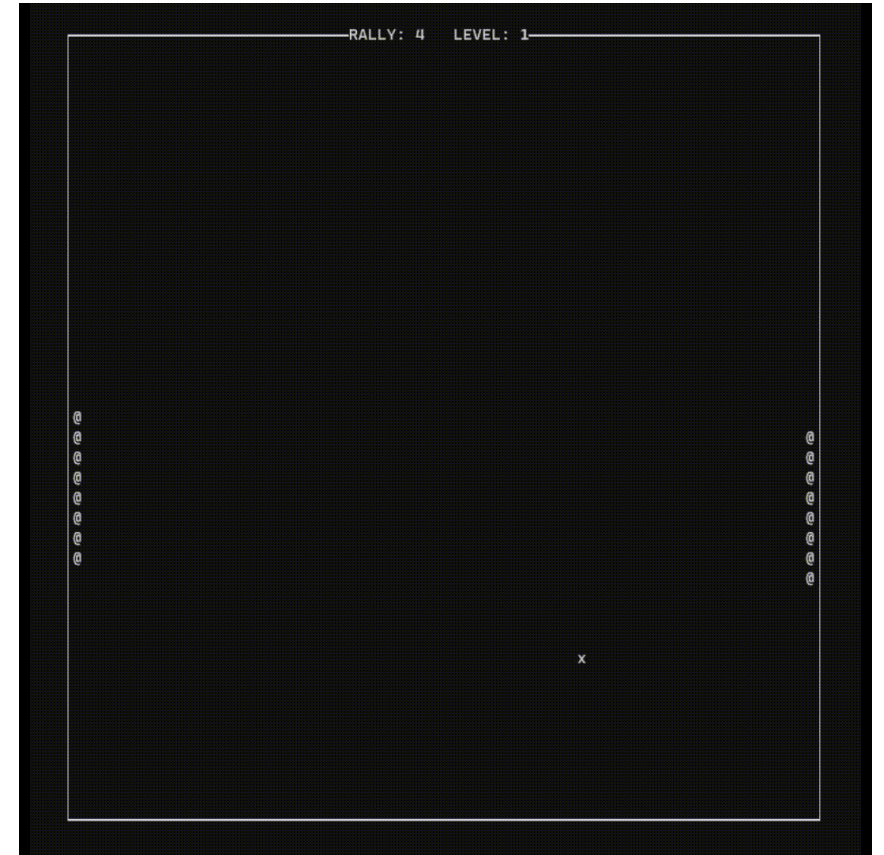


- 공 정보




# 게임 설명 및 규칙

- 사용자는 공이 본인 벽에 맞지 않게 하는 것이 목표이다.
- 방향 키 (위, 아래)를 사용해서 발판을 이동 시켜 공을 막을 수 있다
- 레벨 1~3 까지 있으며, 각 레벨 숫자는 공의 개수를 의미한다.
  - 레벨 1: 랠리 < 5
  - 레벨 2:  $5 \leq \text{랠리} < 10$
  - 레벨 3:  $10 \leq \text{랠리}$
- 공이 사용자 발판 중간에 맞을 때 속도는 1
- 공이 사용자 발판의 끝 부분에 맞을 때 속도는 2



# 게임 설명 및 규칙

- 게임이 종료된 후에 사용자가 r를 클릭하여 게임을 재시작 할 수 있다



Game Over!  
Winner: Client #1  
Press r to replay.

# 개발 방법

- Boost Asio
  - TCP 네트워크 API들을 사용 (read, write)
- Ncurses
  - 사용자 키보드 입력 정보를 받고
  - 터미널 창에 window를 생성하여 x와 y좌표로 character 출력하여 frame 단위로 GUI 구현



# 역할 분담

- 안예온
  - 네트워킹 송신 및 수신 검증/구현
- 양희찬
  - 게임 logics 구현
  - Ncurses 활용 출력 구현
- 유현도
  - Multi-thread 검증/구현

# Appendix

평가에 사용

# 자세한 설계 내용

---

## Server (Admin)

---

기본 window (box) 정보 (송신)

공의 정보 (송신)

Player 1 정보 (송신)

Player 2 정보 (송신)

Player 1 & 2 입력 값 (수신)

---

---

## Client (Player 1)

---

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

---

---

## Client (Player 2)

---

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

---



# 자세한 설계 내용: Client

---

## Server (Admin)

기본 window (box) 정보 (송신)

공의 정보 (송신)

Player 1 정보 (송신)

Player 2 정보 (송신)

Player 1 & 2 입력 값 (수신)

---

---

## Client (Player 1)

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

---

---

## Client (Player 2)

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

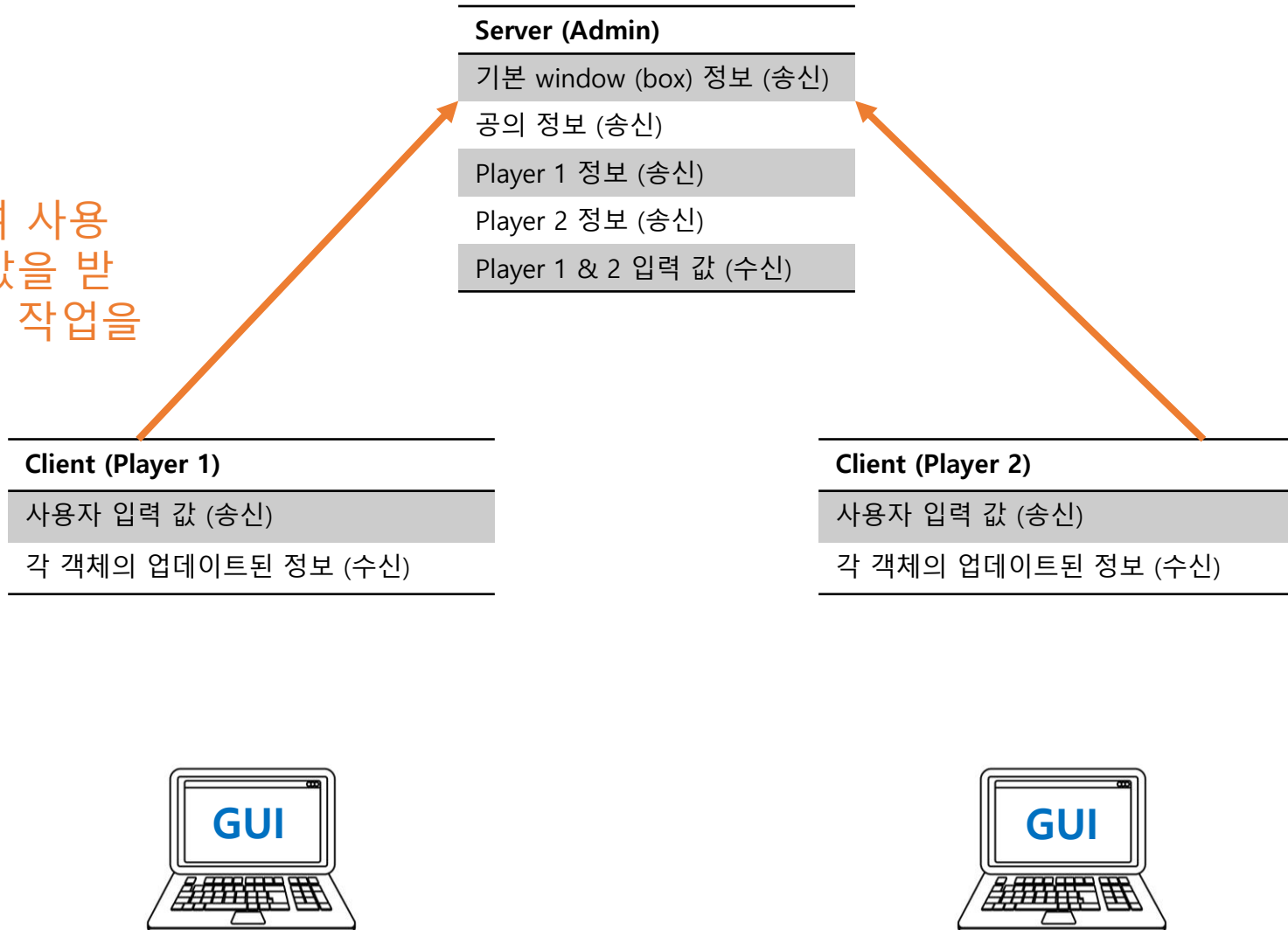
---



# 자세한 설계 내용: Client

## Sending Thread

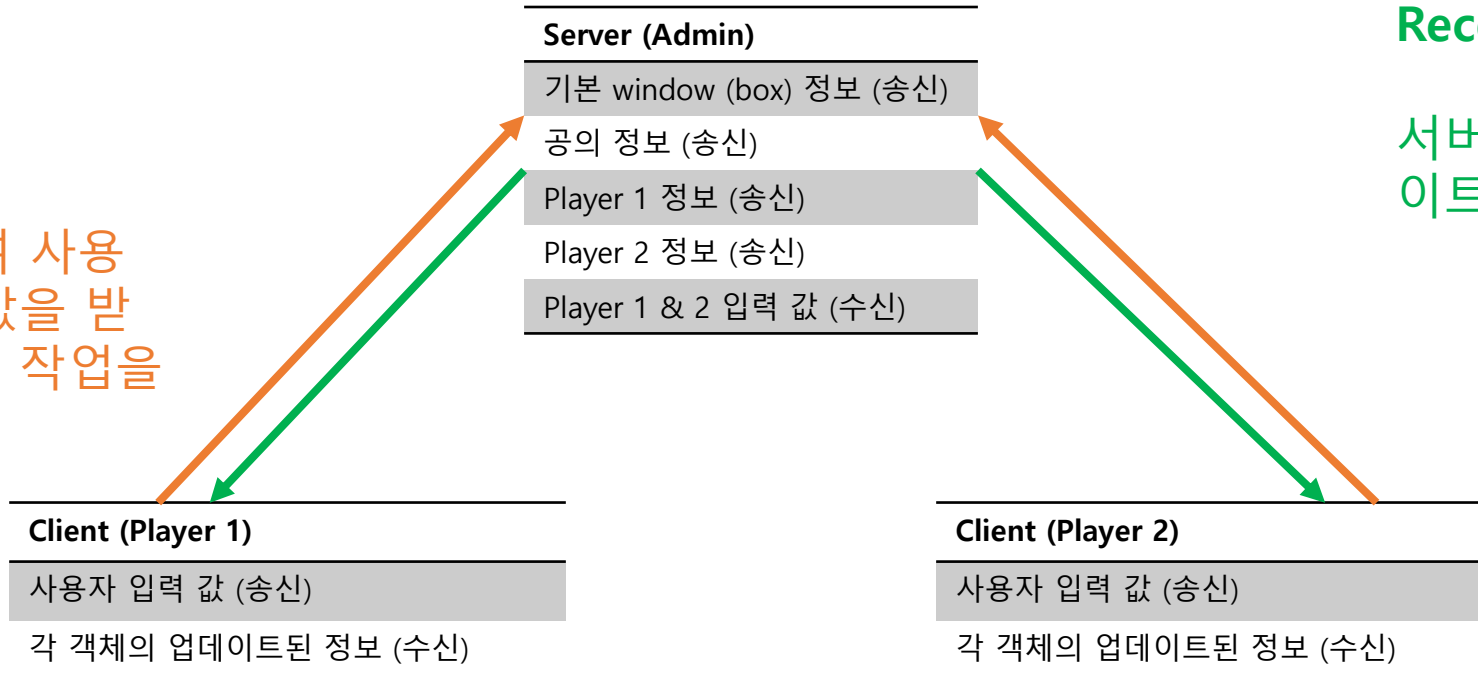
Ncurses를 활용하여 사용자의 키보드 입력 값을 받고 서버에 수신하는 작업을 반복한다.



# 자세한 설계 내용: Client

## Sending Thread

Ncurses를 활용하여 사용자의 키보드 입력 값을 받고 서버에 수신하는 작업을 반복한다.



## Receiving Thread

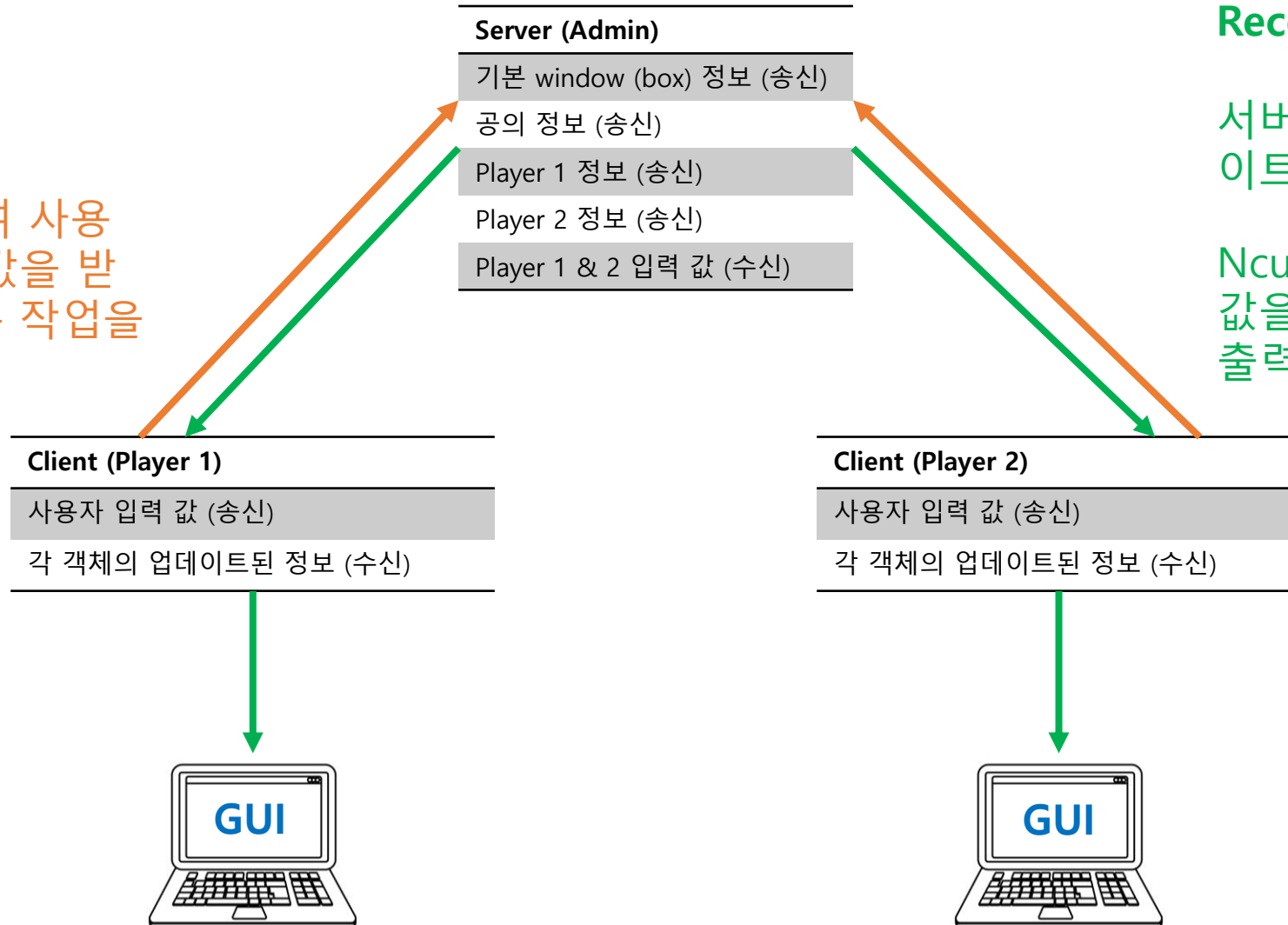
서버에서 각 객체들의 업데이트된 정보를 송신한다.



# 자세한 설계 내용: Client

## Sending Thread

Ncurses를 활용하여 사용자의 키보드 입력 값을 받고 서버에 수신하는 작업을 반복한다.



## Receiving Thread

서버에서 각 객체들의 업데이트된 정보를 송신한다.

Ncurses를 활용하여 해당 값을 가지고 화면을 다시 출력한다.

# 자세한 설계 내용: Server

---

## Server (Admin)

기본 window (box) 정보 (송신)

공의 정보 (송신)

Player 1 정보 (송신)

Player 2 정보 (송신)

Player 1 & 2 입력 값 (수신)

---

---

## Client (Player 1)

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

---

---

## Client (Player 2)

사용자 입력 값 (송신)

각 객체의 업데이트된 정보 (수신)

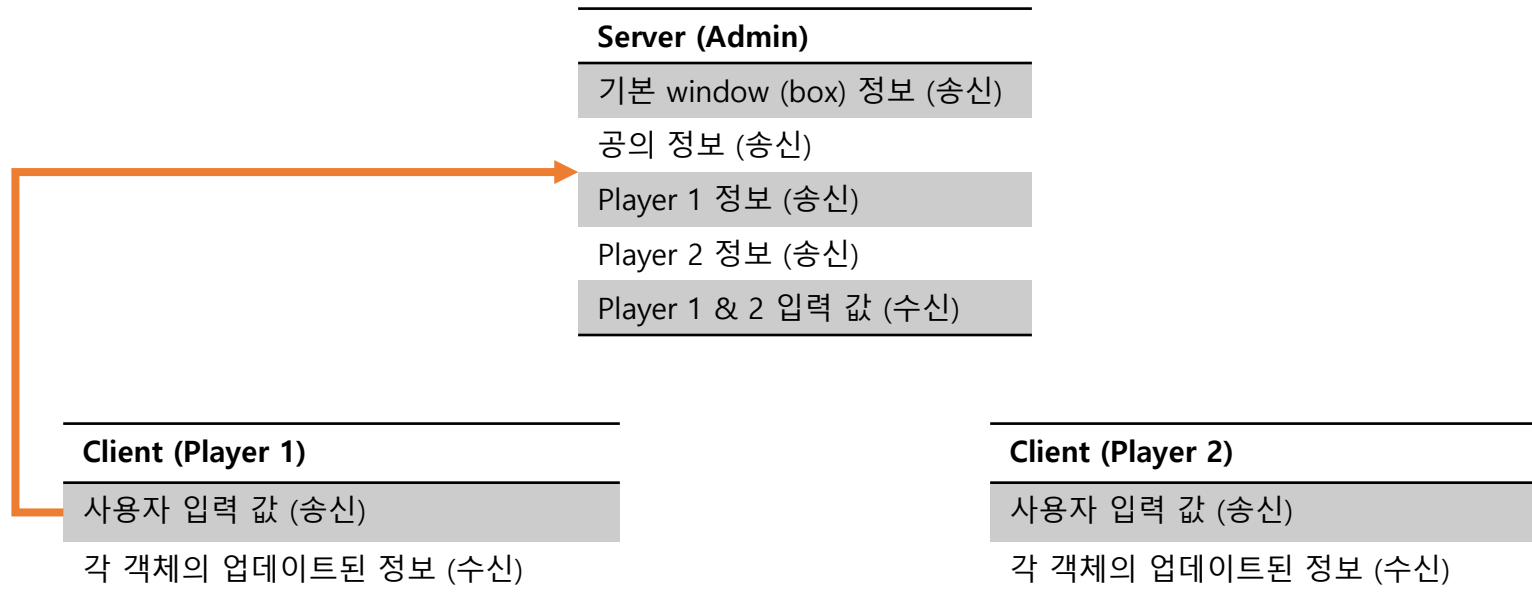
---



# 자세한 설계 내용: Server

## P1 Thread

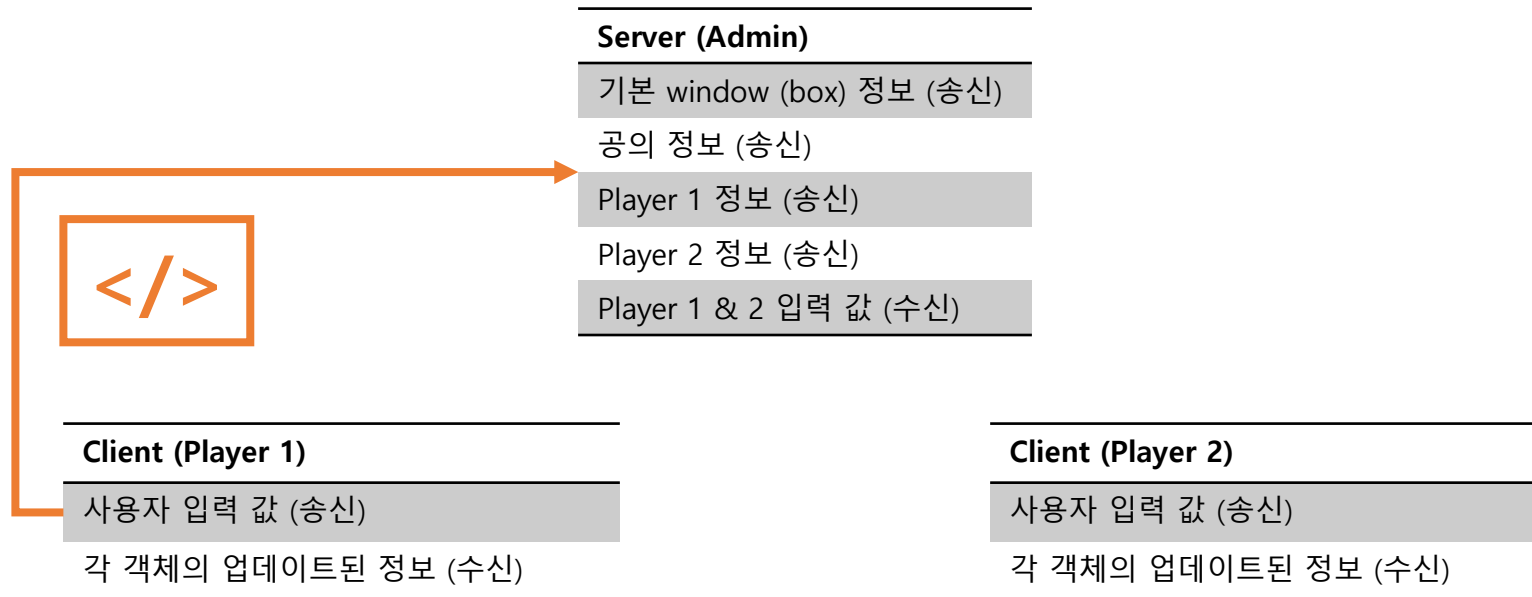
사용자 입력 값을 수신한다.



# 자세한 설계 내용: Server

## P1 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.



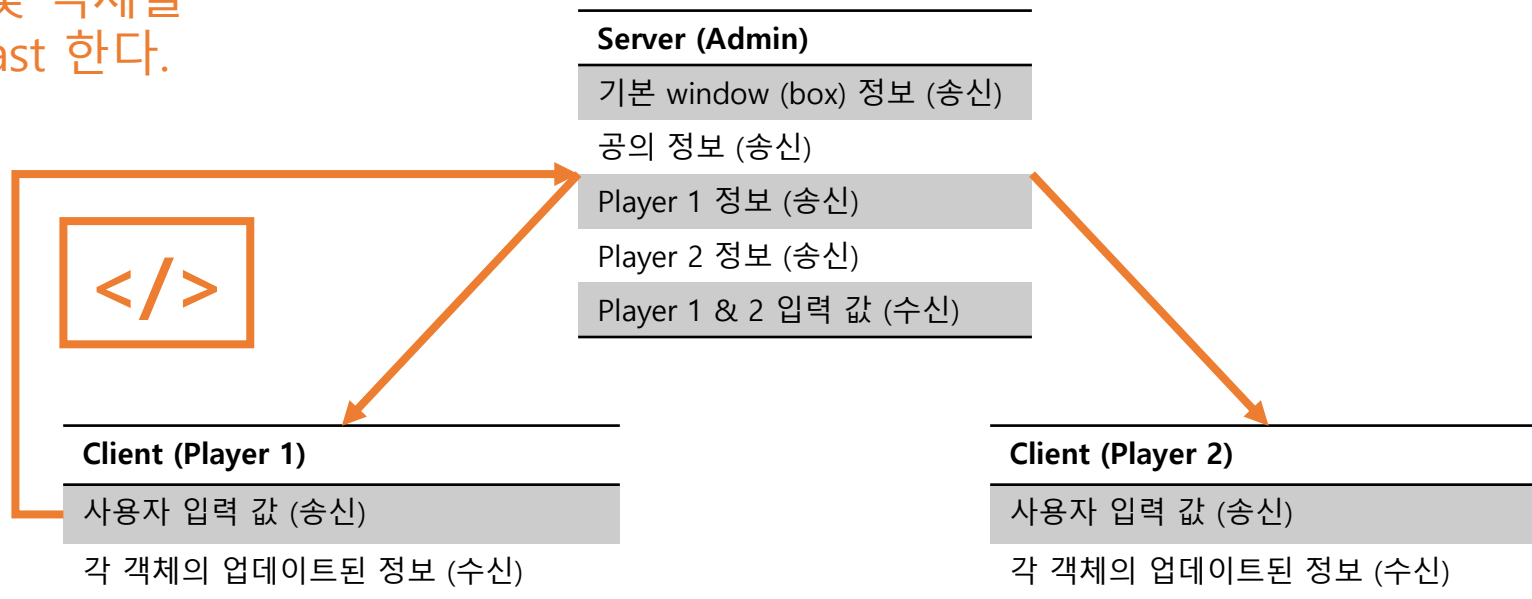
# 자세한 설계 내용: Server

## P1 Thread

사용자 입력 값을 수신한다.

해당 사용자의 발판 위치를  
업데이트 한다.

업데이트된 정보 및 객체들  
의 정보를 broadcast 한다.



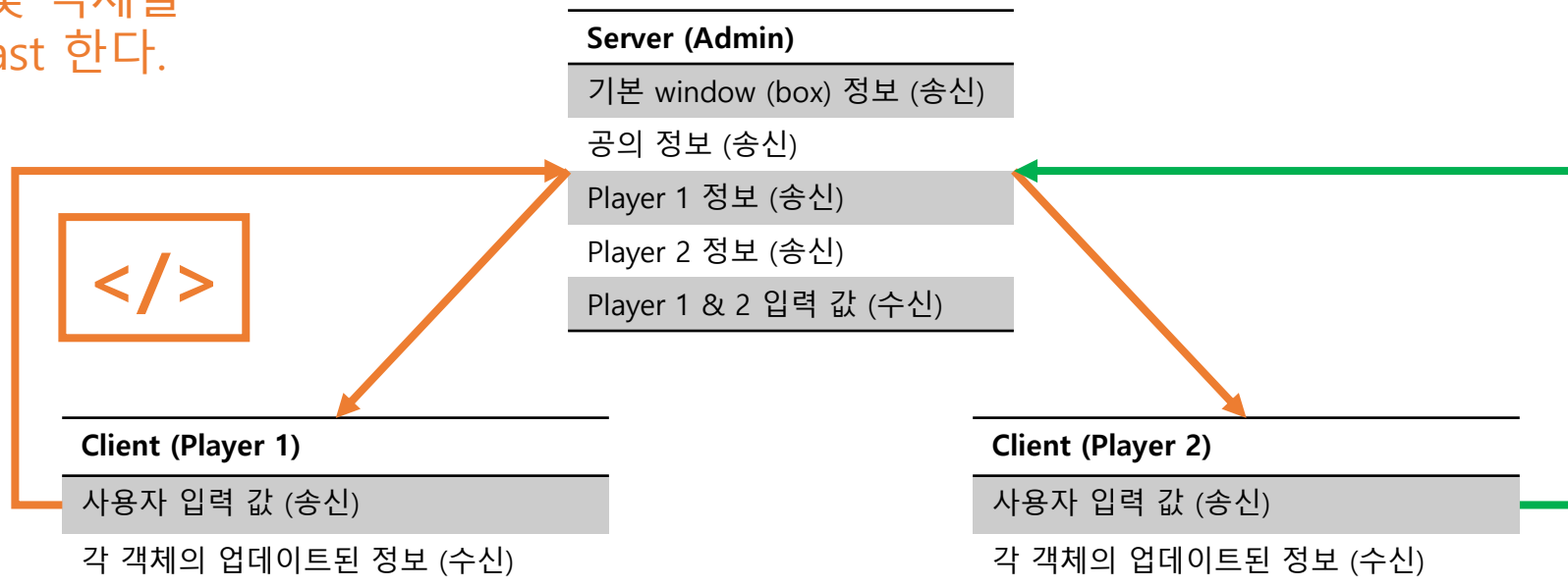
# 자세한 설계 내용: Server

## P1 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.  
업데이트된 정보 및 객체들  
의 정보를 broadcast 한다.

## P2 Thread

사용자 입력 값을 수신한다.



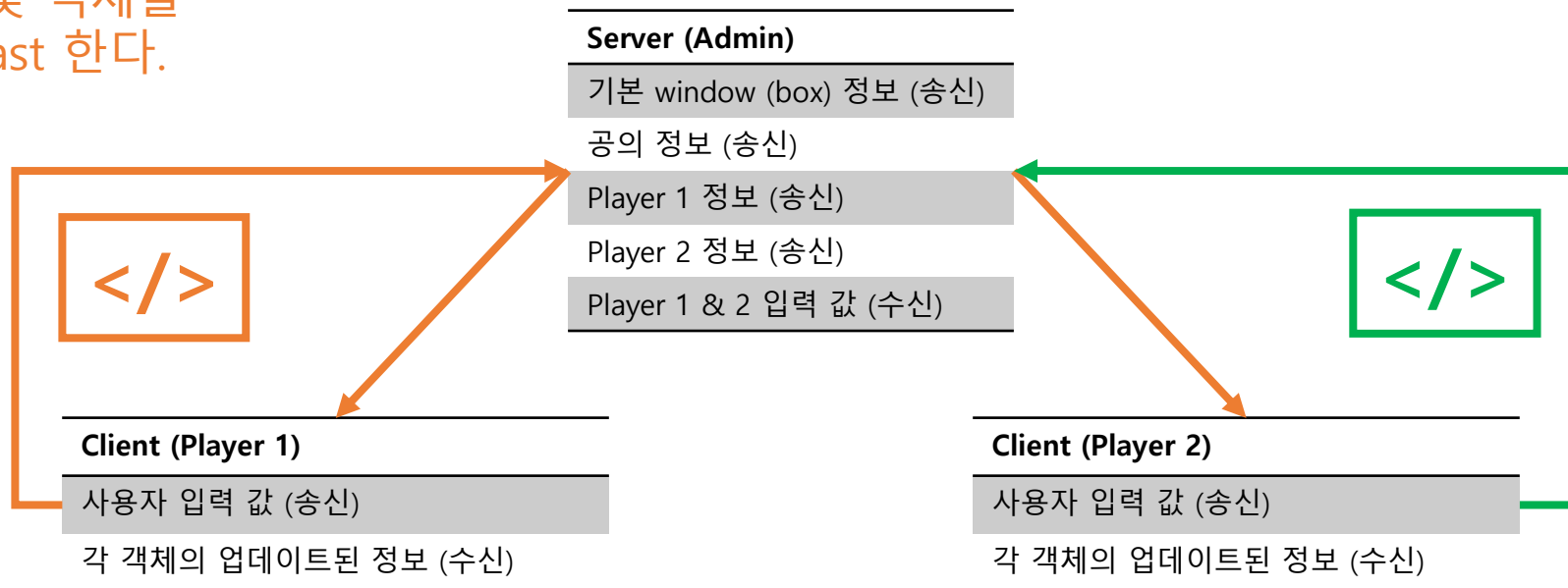
# 자세한 설계 내용: Server

## P1 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.  
업데이트된 정보 및 객체들  
의 정보를 broadcast 한다.

## P2 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.



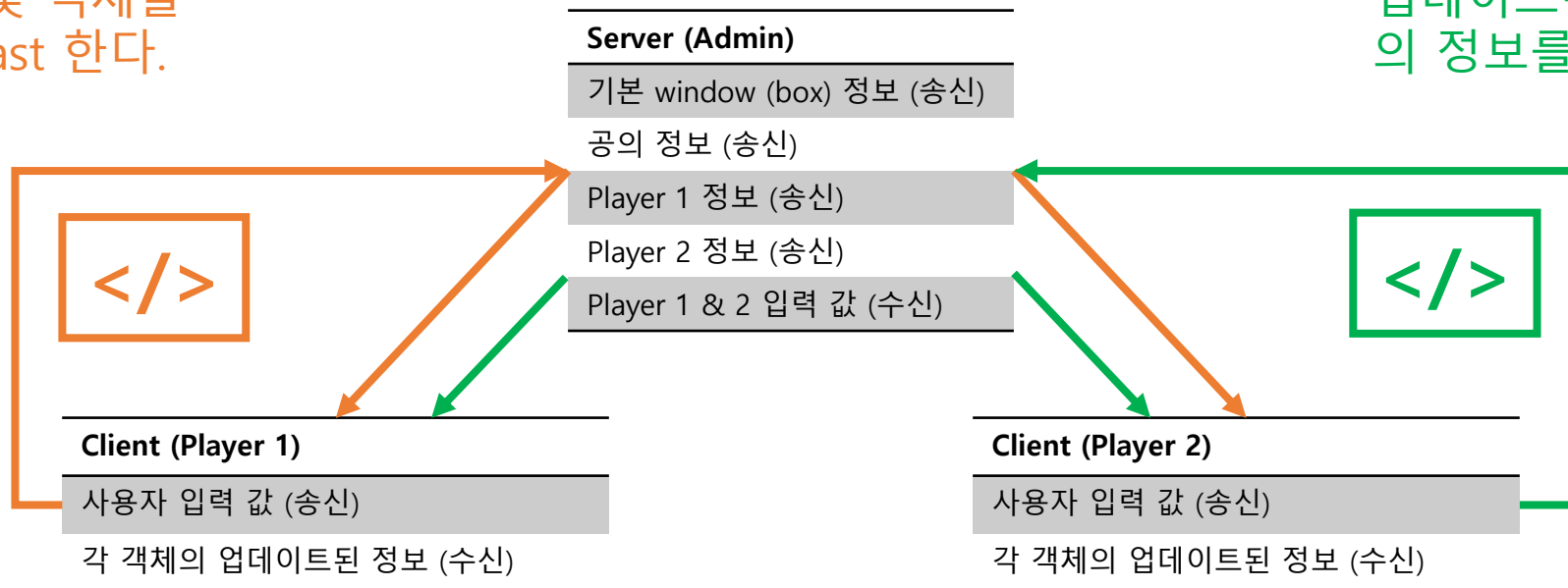
# 자세한 설계 내용: Server

## P1 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.  
업데이트된 정보 및 객체들  
의 정보를 broadcast 한다.

## P2 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를  
업데이트 한다.  
업데이트된 정보 및 객체들  
의 정보를 broadcast 한다.



# 자세한 설계 내용: Server

## P1 Thread

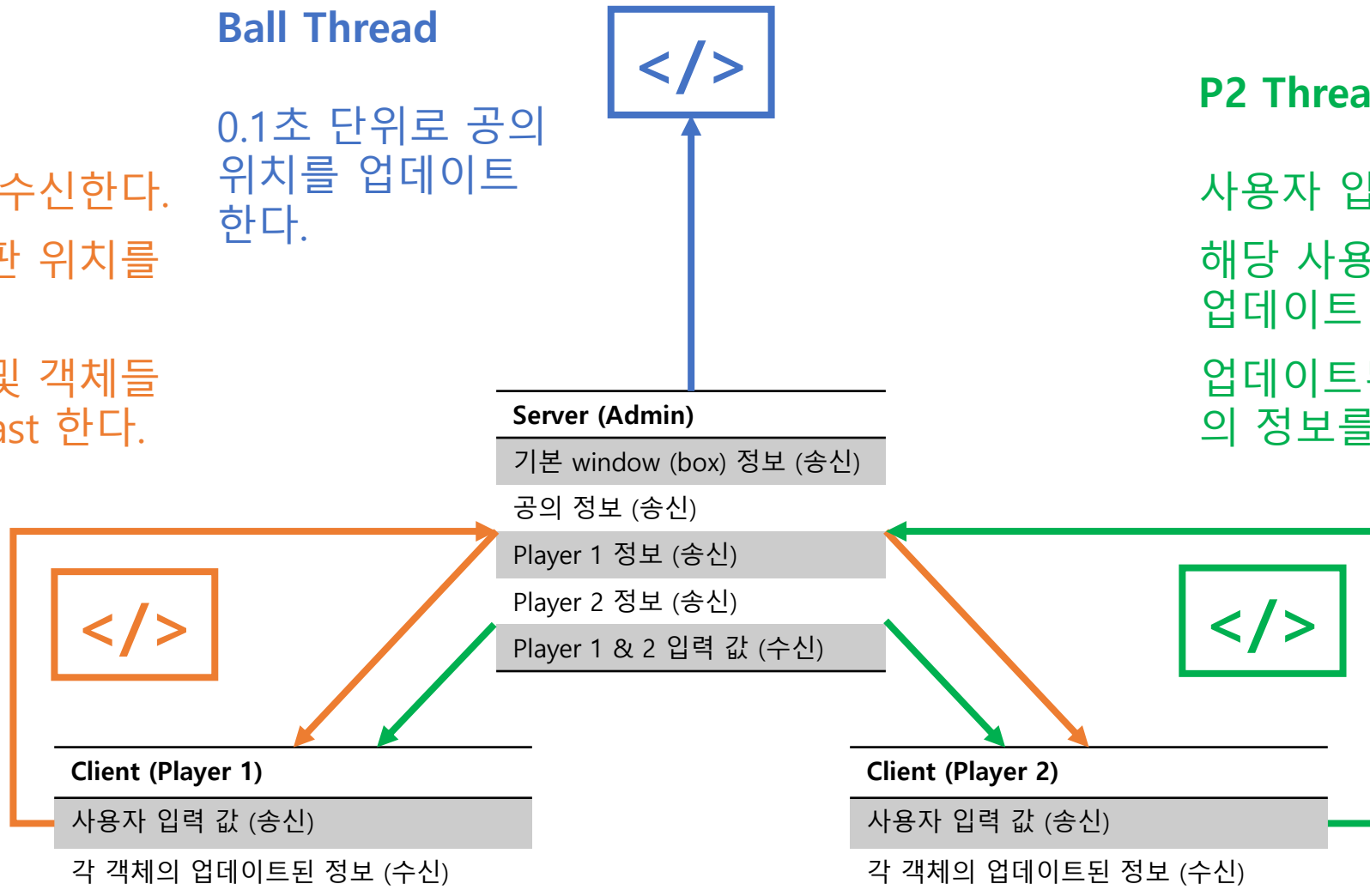
사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를 업데이트 한다.  
업데이트된 정보 및 객체들의 정보를 broadcast 한다.

## Ball Thread

0.1초 단위로 공의 위치를 업데이트 한다.

## P2 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를 업데이트 한다.  
업데이트된 정보 및 객체들의 정보를 broadcast 한다.



# 자세한 설계 내용: Server

## P1 Thread

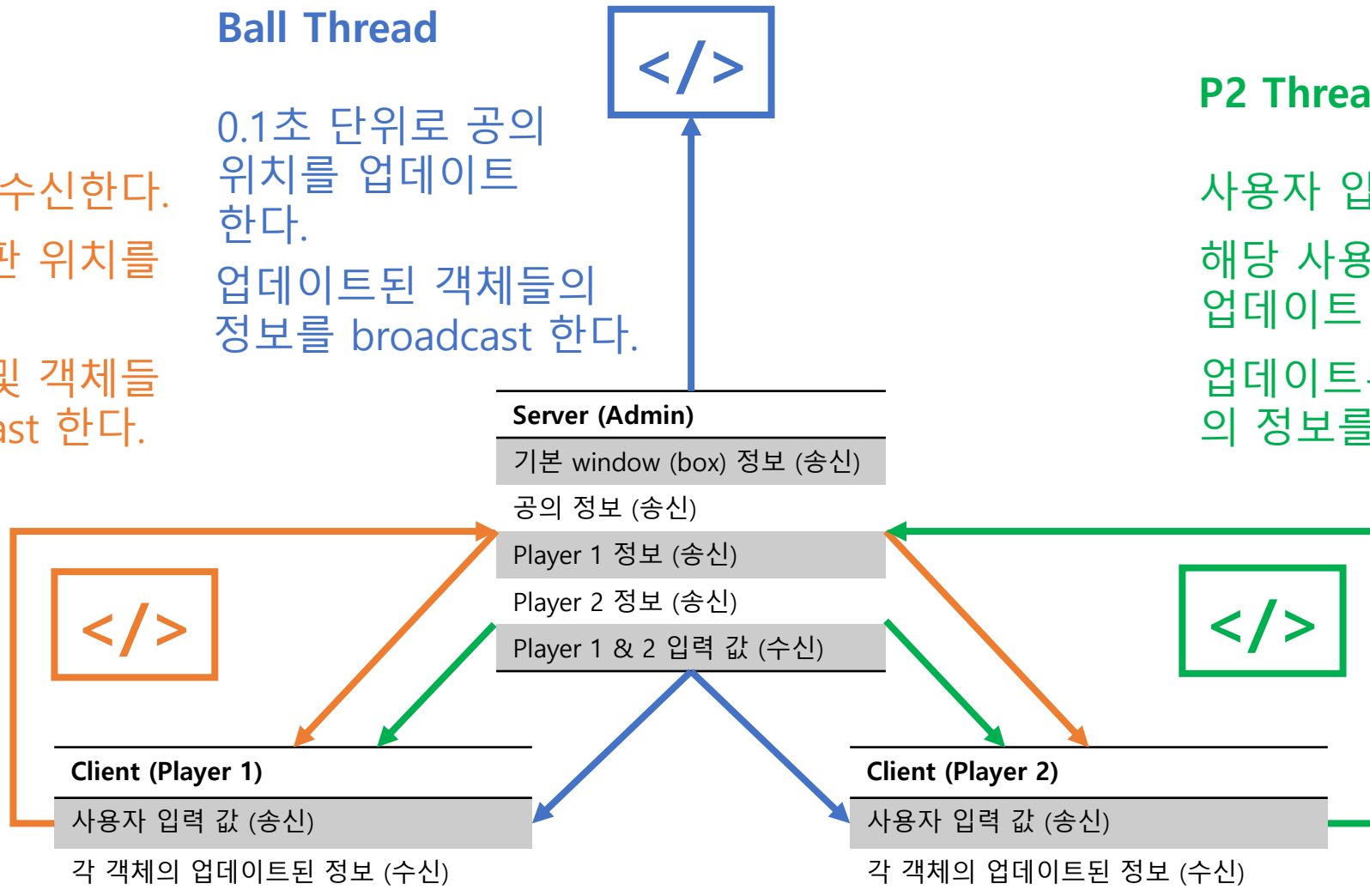
사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를 업데이트 한다.  
업데이트된 정보 및 객체들의 정보를 broadcast 한다.

## Ball Thread

0.1초 단위로 공의 위치를 업데이트 한다.  
업데이트된 객체들의 정보를 broadcast 한다.

## P2 Thread

사용자 입력 값을 수신한다.  
해당 사용자의 발판 위치를 업데이트 한다.  
업데이트된 정보 및 객체들의 정보를 broadcast 한다.

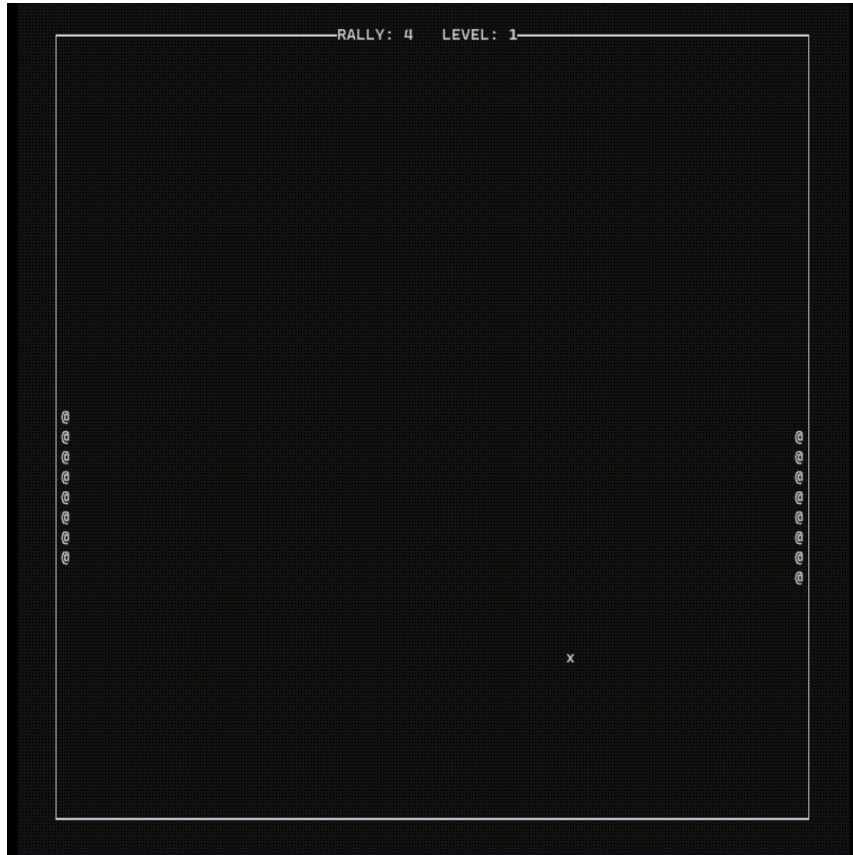




# Github 주소

<https://github.com/yheechan/Network-Based-Multiplayer-PingPong-Game.git>

# 시현 화면 캡처



# 감사합니다.

Team1: 4 to 6